

# Order Statistics Based List Decoding Techniques for Linear Binary Block Codes

Saif E. A. Alnawayseh, and Pavel Loskot, *Member, IEEE*

## Abstract

The order statistics based list decoding techniques for linear binary block codes of small to medium block length are investigated. The construction of the list of the test error patterns is considered. The original order statistics decoding is generalized by assuming segmentation of the most reliable independent positions of the received bits. The segmentation is shown to overcome several drawbacks of the original order statistics decoding. The complexity of the order statistics based decoding is further reduced by assuming a partial ordering of the received bits in order to avoid the complex Gauss elimination. The probability of the test error patterns in the decoding list is derived. The bit error rate performance and the decoding complexity trade-off of the proposed decoding algorithms is studied by computer simulations. Numerical examples show that, in some cases, the proposed decoding schemes are superior to the original order statistics decoding in terms of both the bit error rate performance as well as the decoding complexity.

## Index Terms

Decoding, fading, linear code, performance evaluation.

The authors are with College of Engineering, Swansea University, Singleton Park, Swansea SA2 8PP, United Kingdom (email: [491959,p.loskot}@swan.ac.uk](mailto:{491959,p.loskot}@swan.ac.uk))

Corresponding author: Pavel Loskot, phone: +44 1792 602619, fax: +44 1792 295676

This work was presented in part at the *IEEE Wireless Communications & Signal Processing Conference (WCSP)*, Nanjing, China, Nov. 2009.

## I. INTRODUCTION

A major difficulty in employing forward error correction (FEC) coding is the implementation complexity especially of the decoding at the receiver and the associated decoding latency for long codewords. Correspondingly, the FEC coding is often designed to trade-off the bit error rate (BER) with the decoding complexity and latency. Many universal decoding algorithms have been proposed for the decoding linear binary block codes [1]. The decoding algorithms in [3]–[11] are based on the testing and re-encoding of the information bits as initially considered by Dorsch in [2]. In particular, a list of the likely transmitted codewords is generated using the reliabilities of the received bits, and then, the most likely codeword is selected from this list. The list of the likely transmitted codewords can be constructed from a set of the test error patterns. The test error patterns can be predefined as in [3] and [4] and in this paper, predefined and optimized for the channel statistics as in [5], or defined adaptively for a particular received sequence as suggested in [6]. The complexity of the list decoding can be further reduced by the skipping and stopping rules as shown, for example, in [3] and [4].

Among numerous variants of the list decoding techniques, the order statistics decoding (OSD) is well-known [3], [4]. The structural properties of the FEC code are utilized to reduce the OSD complexity in [7]. The achievable coding gain of the OSD is improved by considering the multiple information sets in [8]. The decoding proposed in [9] exploits randomly generated biases to present the decoder with the multiple received soft-decision values. The sort and match decoding of [10] divides the received sequence into two disjoint segments. The list decoding is then performed for each of the two segments independently, and the two lists are combined using the sort and match algorithm to decide on the most likely transmitted codeword. The box and match decoding strategy is developed in [11]. An alternative approach to the soft-decision decoding of linear binary block codes relies on the sphere decoding techniques [12], [13]. For example, the input sphere decoder (ISD) discussed in this paper can be considered to be a trivial sphere decoding algorithm.

In this paper, we investigate the OSD-based decoding strategies for linear binary block codes. Our aim is to obtain low-complexity decoding schemes that provide sufficiently large or valuable coding gains, and most importantly, that are well-suited for implementation in communication systems with limited hardware resources, e.g., at nodes of the wireless sensor network. We

modify the original OSD by considering the disjoint segments of the most reliable independent positions (MRIPs). The segmentation of the MRIPs creates flexibility that can be exploited to fine tune a trade-off between the BER performance and the decoding complexity. Thus, the original OSD can be considered to be a special case of the segmentation-based OSD having only one segment corresponding to the MRIPs. Since the complexity of obtaining a row echelon form of the generator matrix for every received codeword represents a significant part of the overall decoding complexity, we examine a partial-order statistics decoding (POSD) when only the systematic part of the received codeword is ordered.

This paper is organized as follows. System model is described in Section II. Construction of the list of test error patterns is investigated in Section III. The list decoding algorithms are developed in Section IV. The performance analysis is considered in Section V. Numerical examples to compare the BER performance and the decoding complexity of the proposed decoding schemes are presented in Section VI. Finally, conclusions are given in Section VII.

## II. SYSTEM MODEL

Consider transmission of codewords of a linear binary block code  $\mathcal{C}$  over an additive white Gaussian noise (AWGN) channel with Rayleigh fading. The code  $\mathcal{C}$ , denoted as  $(N, K, d_{\min})$ , has block length  $N$ , dimension  $K$ , and the minimum Hamming distance between any two codewords  $d_{\min}$ . Binary codewords  $\mathbf{c} \in \mathbb{Z}_2^N$  where  $\mathbb{Z}_2 = \{0, 1\}$  are generated from a vector of information bits  $\mathbf{u} \in \mathbb{Z}_2^K$  using the generator matrix  $\mathbf{G} \in \mathbb{Z}_2^{K \times N}$ , i.e.,  $\mathbf{c} = \mathbf{u}\mathbf{G}$ , and all binary operations are considered over a Galois field GF(2). If the code  $\mathcal{C}$  is systematic, the generator matrix has the form,  $\mathbf{G} = [\mathbf{I} \ \mathbf{P}]$ , where  $\mathbf{I}$  is the  $K \times K$  identity matrix, and  $\mathbf{P} \in \mathbb{Z}_2^{K \times (N-K)}$  is the matrix of parity checks. The codeword  $\mathbf{c}$  is mapped to a binary phase shift keying (BPSK) sequence  $\mathbf{x} \in \{+1, -1\}^N$  before transmission using a mapping,  $x_i = \mathcal{M}(c_i) = (-1)^{c_i}$ , for  $i = 1, 2, \dots, N$ . Assuming bits  $u_i$  and  $u_j$ , the mapping  $\mathcal{M}$  has the property,

$$\mathcal{M}(u_i \oplus u_j) = \mathcal{M}(u_i) \mathcal{M}(u_j) \quad (1)$$

where  $\oplus$  denotes the modulo 2 addition. The encoded bit  $c_i$  can be recovered from the symbol  $x_i$  using the inverse mapping,  $c_i = \mathcal{M}^{-1}(x_i) = (1 - x_i)/2$ . For brevity, we also use the notation,  $\mathbf{x} = \mathcal{M}(\mathbf{c})$  and  $\mathbf{c} = \mathcal{M}^{-1}(\mathbf{x})$ , to denote the component-wise modulation mapping and de-mapping, respectively. The code  $\mathcal{C}$  is assumed to have equally probable values of the encoded

bits, i.e., the probability,  $\Pr\{c_i = 0\} = \Pr\{c_i = 1\} = 1/2$ , for  $i = 1, 2, \dots, N$ . Consequently, all the codewords are transmitted with the equal probability, i.e.,  $\Pr\{\mathbf{c}\} = 2^{-K}$  for  $\forall \mathbf{c} \in \mathcal{C}$ .

The signal at the output of the matched filter at the receiver can be written as,

$$y_i = h_i x_i + w_i$$

where the frequency non-selective channel fading coefficients  $h_i$  as well as the AWGN samples  $w_i$  are mutually uncorrelated zero-mean circularly symmetric complex Gaussian random variables. The variance of  $h_i$  is unity, i.e.,  $\text{E}[|h_i|^2] = 1$  where  $\text{E}[\cdot]$  is expectation, and  $|\cdot|$  denotes the absolute value. The samples  $w_i$  have the variance,  $\text{E}[|w_i|^2] = (R\gamma_c)^{-1}$ , where  $R = K/N$  is the coding rate of  $\mathcal{C}$ , and  $\gamma_c$  is the signal-to-noise ratio (SNR) per transmitted encoded binary symbol. The covariance,  $\text{E}[h_i h_j^*] = 0$  for  $i \neq j$ , where  $(\cdot)^*$  denotes the complex conjugate, corresponds to the case of a fast fading channel with ideal interleaving and deinterleaving. For a slowly block-fading channel, the covariance,  $\text{E}[h_i h_j^*] = 1$  for  $\forall i, j = 1, 2, \dots, N$ , and the fading coefficients are uncorrelated between transmissions of adjacent codewords.

In general, denote as  $f(\cdot)$  the probability density function (PDF) of a random variable. The reliability  $r_i$  of the received signal  $y_i$  corresponds to a ratio of the conditional PDFs of  $y_i$  [14], i.e.,

$$\frac{f(y_i|x_i = +1, h_i)}{f(y_i|x_i = -1, h_i)} \propto \text{Re}\{h_i^* y_i\} = r_i$$

since the PDF  $f(y_i|x_i, h_i)$  is conditionally Gaussian. Thus, the reliability  $r_i$  can be written as,

$$r_i = \text{Re}\{h_i\} \text{Re}\{y_i\} + \text{Im}\{h_i\} \text{Im}\{y_i\} = |h_i|^2 x_i + |h_i| w_i.$$

The bit-by-bit quantized (i.e., hard) decisions are then defined as,

$$\hat{c}_i = \mathcal{M}^{-1}(\text{sign}(r_i))$$

where  $\text{sign}(\cdot)$  denotes the sign of a real number.

More importantly, even though the primary metric of our interest is the BER performance of the code  $\mathcal{C}$ , it is mathematically more convenient to obtain and analyze the list decoding algorithms assuming the probability of codeword error. Thus, we assume that the list decoding with a given decoding complexity obtained for the probability of codeword error will also have a good BER performance. The maximum likelihood (ML) decoder minimizing the probability

of codeword error provides the decision  $\hat{\mathbf{c}}_{\text{ML}}$  on the most likely transmitted codeword, i.e.,

$$\begin{aligned}\hat{\mathbf{c}}_{\text{ML}} &= \underset{\mathbf{c} \in \mathcal{C}: \mathbf{x} = \mathcal{M}(\mathbf{c})}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{h} \odot \mathbf{x}\|^2 \\ &= \underset{\mathbf{c} \in \mathcal{C}}{\operatorname{argmax}} \sum_{i=1}^N \operatorname{Re}\{y_i h_i^* x_i^*\} \stackrel{\text{BPSK}}{=} \underset{\mathbf{c} \in \mathcal{C}: \mathbf{x} = \mathcal{M}(\mathbf{c})}{\operatorname{argmax}} \mathbf{r} \cdot \mathbf{x}\end{aligned}\quad (2)$$

where  $\mathbf{y}$ ,  $\mathbf{h}$ ,  $\mathbf{x}$ , and  $\mathbf{r}$  denote the  $N$ -dimensional row vectors of the received signals  $y_i$ , the channel coefficients  $h_i$ , the transmitted symbols  $x_i$ , and the reliabilities  $r_i$  within one codeword, respectively,  $\|\cdot\|$  is the Euclidean norm of a vector,  $\odot$  is the component-wise (Hadamard) product of vectors, and the binary operator  $\cdot$  is used to denote the dot-product of vectors. The codewords  $\mathbf{c} \in \mathcal{C}$  used in (2) to find the maximum or the minimum value of the ML metric are often referred to as the test codewords. In the following subsection, we investigate the soft-decision decoding algorithms with low implementation complexity to replace the computationally demanding ML decoding (2).

### A. List Decoding

We investigate the list-based decoding algorithms. For simplicity, we assume binary block codes that are linear and systematic [15]. We note that whereas the extension of the list-based decoding algorithms to non-systematic codes is straightforward, the list based decoding of non-linear codes is complicated by the fact that the list of the test codewords is, in general, dependent on the received sequence. The decoding (time) complexity  $O$  of the list decoding algorithms can be measured as the list size given by the number of the test codewords that are examined in the decoding process. Thus, the ML decoding (2) has the complexity,  $O_{\text{ML}} = 2^K$ , which is prohibitive for larger values of  $K$ . Among the practical list-based decoding algorithms with the acceptable decoding complexity, we investigate the order statistics decoding (OSD) [3] based list decoding algorithms for soft-decision decoding of linear binary block codes.

The OSD decoding resumes by reordering the received sequence of reliabilities as,

$$|\tilde{r}'_1| \geq |\tilde{r}'_2| \geq \cdots \geq |\tilde{r}'_N| \quad (3)$$

where the tilde is used to denote the ordering. This ordering of the reliabilities defines a permutation,  $\lambda'$ , i.e.,

$$\tilde{\mathbf{r}}' = \lambda'[\mathbf{r}] = (\tilde{r}'_1, \dots, \tilde{r}'_N).$$

The permutation  $\lambda'$  corresponds to the generator matrix  $\tilde{\mathbf{G}}' = \lambda'[\mathbf{G}]$  having the reordered columns. In order to obtain the most reliable independent positions (MRIPs) for the first  $K$  bits in the codeword, additional swapping of the columns of  $\tilde{\mathbf{G}}'$  may have to be used which corresponds to the permutation  $\lambda''$ , and the generator matrix  $\tilde{\mathbf{G}}'' = \lambda''[\tilde{\mathbf{G}}']$ . The matrix  $\tilde{\mathbf{G}}''$  can be manipulated into a row (or a reduced row) echelon form using the Gauss (or the Gauss-Jordan) elimination. To simplify the notation, let  $\tilde{\mathbf{r}}$  and  $\tilde{\mathbf{G}}$  denote the reordered sequence of the reliabilities  $\mathbf{r}$  and the reordered generator matrix  $\tilde{\mathbf{G}}$  in a row (or a reduced row) echelon form, respectively, after employing the permutations  $\lambda'$  and  $\lambda''$ , to decode the received sequence  $\mathbf{y}$ . Thus, for  $i \geq j$ , the reordered sequence  $\tilde{\mathbf{r}}$  has elements,  $|\tilde{r}_i| \geq |\tilde{r}_j|$ , for  $i, j = 1, \dots, K$ , and for  $i, j = K+1, \dots, N$ .

The complexity of the ML decoding (2) of the received sequence  $\mathbf{y}$  can be reduced by assuming a list of the  $L$  test codewords, so that  $L \ll 2^K$ . Denote such a list of the test codewords of cardinality  $L$  generated by the matrix  $\tilde{\mathbf{G}}$  as,  $\mathcal{E}_L = \{\mathbf{e}_0, \mathbf{e}_1, \dots, \mathbf{e}_{L-1}\}$ , and let  $\mathbf{e}_0 = \mathbf{0}$  be the all-zero codeword. Then, the list decoding of  $\mathbf{y}$  is defined as,

$$\hat{\mathbf{c}} = \underset{\mathbf{e} \in \mathcal{E}_L: \mathbf{x} = \mathcal{M}(\hat{\mathbf{c}}_0 \oplus \mathbf{e})}{\operatorname{argmax}} \tilde{\mathbf{r}} \cdot \mathbf{x} \quad (4)$$

where the systematic part of the codeword  $\hat{\mathbf{c}}_0$  is given by the hard-decision decoded bits at the MRIPs. The decoding step to obtain the decision  $\hat{\mathbf{c}}_0$  is referred to as the 0-th order OSD reprocessing in [3]. In addition, due to linearity of  $\mathcal{C}$ , we have that  $(\mathbf{c}_0 \oplus \mathbf{e}) \in \mathcal{C}$ , and thus, the test codewords  $\mathbf{e} \in \mathcal{E}_L$  can be also referred to as the test error patterns in the decoding (4). Using the property (1), we can rewrite the decoding (4) as,

$$\hat{\mathbf{c}} = \underset{\mathbf{e} \in \mathcal{E}_L}{\operatorname{argmax}} \tilde{\mathbf{r}} \cdot \hat{\mathbf{x}}_0 \cdot \mathcal{M}(\mathbf{e}) = \underset{\mathbf{e} \in \mathcal{E}_L}{\operatorname{argmax}} \tilde{\mathbf{r}}_0 \cdot \mathcal{M}(\mathbf{e}) \quad (5)$$

where we denoted  $\hat{\mathbf{x}}_0 = \mathcal{M}(\hat{\mathbf{c}}_0)$  and  $\tilde{\mathbf{r}}_0 = \tilde{\mathbf{r}} \odot \hat{\mathbf{x}}_0$ . The system model employing the list decoding (5) is illustrated in Fig. 1. More importantly, as indicated in Fig. 1, the system model can be represented as an equivalent channel with the binary vector input  $\mathbf{c}$  and the vector soft-output  $\tilde{\mathbf{r}}_0$ .

### III. LIST SELECTION

The selection of the test error patterns  $\mathbf{e}$  to the list  $\mathcal{E}_L$  as well as the list size  $L$  have a dominant effect upon the probability of incorrect codeword decision by the list decoding. Denote

such probability of codeword error as  $P_e$ , and let  $\mathbf{c}_{Tx}$  be the transmitted codeword. In [12], the probability  $P_e$  is expanded as,

$$\begin{aligned} P_e &= \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\} \Pr\{\hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\} \\ &\quad + \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\} \Pr\{\hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\} \end{aligned}$$

where the decision  $\hat{\mathbf{c}}$  is obtained by the decoding (5), and the condition,  $\hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}$ , is true provided that the vectors  $\hat{\mathbf{c}}_{ML}$  and  $\mathbf{c}_{Tx}$  differ in at least one component, i.e.,  $\hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}$  if and only if all the components of the vectors are equal. Since, for any list  $\mathcal{E}_L$ , the probability,  $\Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\} = 1$ , and usually, the probability,  $\Pr\{\hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\}$  is close to 1,  $P_e$  can be tightly upper-bounded as,

$$P_e \leq \Pr\{\hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\} + \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\}. \quad (6)$$

The first term on the right hand side of (6) is the codeword error probability of the ML decoding, and the second term is the conditional codeword error probability of the list decoding. The probability,  $\Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | \hat{\mathbf{c}}_{ML} = \mathbf{c}_{Tx}\}$ , is decreasing with the list size. In the limit of the maximum list size when the list decoding becomes the ML decoding, the bound (6) becomes,  $P_e = \Pr\{\hat{\mathbf{c}}_{ML} \neq \mathbf{c}_{Tx}\}$ . The bound (6) is particularly useful to analyze the performance of the list decoding (5). However, in order to construct the list of the test error patterns, we consider the following expansion of the probability  $P_e$ , i.e.,

$$\begin{aligned} P_e &= \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | (\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\} \Pr\{(\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\} \\ &\quad + \Pr\{\hat{\mathbf{c}} \neq \mathbf{c}_{Tx} | (\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \notin \mathcal{E}_L\} \Pr\{(\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \notin \mathcal{E}_L\} \\ &= 1 - \underbrace{\Pr\{\hat{\mathbf{c}} = \mathbf{c}_{Tx} | (\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\}}_{P_I} \underbrace{\Pr\{(\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \in \mathcal{E}_L\}}_{P_{II}}. \end{aligned}$$

Using (4) and (5), the probability  $P_I$  that the list decoding (5) selects the transmitted codeword provided that such codeword is in the list (more precisely, provided that the error pattern  $\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0$  is in the list) can be expressed as,

$$P_I = \Pr\{\tilde{\mathbf{r}} \cdot \mathcal{M}(\mathbf{c}_{Tx} \oplus \hat{\mathbf{c}}_0) \geq \tilde{\mathbf{r}}_0 \cdot \mathcal{M}(\mathbf{e}), \forall \mathbf{e} \in \mathcal{E}_L\}. \quad (7)$$

The probability (7) decreases with the list size, and, in the limit of the maximum list size  $L = 2^K$ ,  $P_I = 1 - P_e$ . On the other hand, the probability  $P_{II}$  that the transmitted codeword is in the decoding list increases with the list size, and  $P_{II} = 1$ , for  $L = 2^K$ .

Since the coding  $\mathcal{C}$  and the communication channel are linear, then, without loss of generality, we can assume that the all-zero codeword,  $\mathbf{c}_{\text{Tx}} = \mathbf{0}$ , is transmitted. Consequently, given the list decoding complexity  $L$ , the optimum list  $\mathcal{E}_L^*$  minimizing the probability  $P_e$  is constructed as,

$$\mathcal{E}_L^* = \operatorname{argmax}_{\mathcal{E}: |\mathcal{E}|=L} \Pr\{\hat{\mathbf{c}} = \mathbf{0} | \hat{\mathbf{c}}_0 \in \mathcal{E}\} \Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}\} \quad (8)$$

where  $|\mathcal{E}|$  is the cardinality of the test list  $\mathcal{E}$ , and the hard-decision codeword  $\hat{\mathbf{c}}_0 \in \mathcal{C}$  represents the error pattern observed at the receiver after transmission of the codeword  $\mathbf{c}_{\text{Tx}} = \mathbf{0}$ . For a given list of the error patterns  $\mathcal{E}$  in (8), and for the system model in Section II with asymptotically large SNR, the probability  $P_I = \Pr\{\hat{\mathbf{c}} = \mathbf{0} | \hat{\mathbf{c}}_0 \in \mathcal{E}\}$  is dominated by the error events corresponding to the error patterns with the smallest Hamming distances. Since the error patterns are also codewords of  $\mathcal{C}$ , the smallest Hamming distance between any two error patterns in the list  $\mathcal{E}$  is at least  $d_{\min}$ . Assuming that the search in (8) is constrained to the lists  $\mathcal{E}$  having the minimum Hamming distance between any two error patterns given by  $d_{\min}$ , the probability  $P_I$  is approximately constant for all the lists  $\mathcal{E}$ , and we can consider the suboptimum list construction,

$$\mathcal{E}_L = \operatorname{argmax}_{\mathcal{E}: |\mathcal{E}|=L} \Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}\}. \quad (9)$$

The list construction (9) is recursive in its nature, since the list  $\mathcal{E}$  maximizing (9) consists of the  $L$  most probable error patterns. However, in order to achieve a small probability of decoding error  $P_e$  and approach the probability of decoding error,  $\Pr\{\hat{\mathbf{c}}_{\text{ML}} \neq \mathbf{c}_{\text{Tx}}\}$ , of the ML decoding, the list size  $L$  must be large. We can obtain a practical list construction by assuming the  $L$  sufficiently probable error patterns rather than assuming the  $L$  most likely error patterns. We restate Theorem 1 and Theorem 2 in [3] to obtain the likely error patterns and to define the practical list decoding algorithms.

Denote as  $P(i_1, i_2, \dots, i_n)$  the  $n$ -th order joint probability of bit errors at bit positions  $1 \leq i_1 < i_2 < \dots < i_n \leq N$  in the received codeword after the ordering  $\lambda'$  and  $\lambda''$  and before the decoding. Since the test error pattern  $\mathbf{e}$  is a codeword of  $\mathcal{C}$ , the probability  $P(i_1, i_2, \dots, i_n)$ , for  $i_n \leq K$ , is equal to the probability  $\Pr\{\mathbf{e} = \hat{\mathbf{c}}_0\}$  assuming that the  $n$  bit errors occurred during the transmission corresponding to the positions (after the ordering)  $i_1, i_2, \dots, i_n$ . We have the following lemma.

*Lemma 1:* For any bit positions  $\mathcal{I}_1 \subseteq \mathcal{I} \subseteq \{1, 2, \dots, N\}$ ,

$$P(\mathcal{I}) \leq P(\mathcal{I}_1).$$

*Proof:* The lemma is proved by noting that  $P(\mathcal{I}) = P(\mathcal{I}_1, \mathcal{I} \setminus \mathcal{I}_1) = P(\mathcal{I} \setminus \mathcal{I}_1 | \mathcal{I}_1)P(\mathcal{I}_1) \leq \min\{P(\mathcal{I}_1), P(\mathcal{I} \setminus \mathcal{I}_1 | \mathcal{I}_1)\} \leq P(\mathcal{I}_1)$  where  $\mathcal{I} \setminus \mathcal{I}_1$  denotes the difference of the two sets.  $\blacksquare$

Using Lemma 1, we can show, for example, that,  $P(i, j) \leq P(i)$  and  $P(i, j) \leq P(j)$ . We can now restate Theorem 1 and Theorem 2 in [3] as follows.

*Theorem 1:* Assume bit positions  $1 \leq i < j < k \leq N$ , and let the corresponding reliabilities be  $|\tilde{r}_i| \geq |\tilde{r}_j| \geq |\tilde{r}_k|$ . Then, the bit error probabilities,

$$\begin{aligned} P(i) &\leq P(j) \\ P(i, j) &\leq P(i, k). \end{aligned}$$

*Proof:* Without loss of generality, we assume that the symbols  $x_i = -1$ ,  $x_j = -1$  and  $x_k = -1$  have been transmitted. Then, before the decoding, the received bits would be decided erroneously if the reliabilities  $\tilde{r}_i > 0$ ,  $\tilde{r}_j > 0$ , and  $\tilde{r}_k > 0$ . Conditioned on the transmitted symbols, let  $f(\cdot)$  denote the conditional PDF of the ordered reliabilities  $\tilde{r}_i$ ,  $\tilde{r}_j$  and  $\tilde{r}_k$ .

Consider first the inequality  $P(i) \leq P(j)$ . Since, for  $\tilde{r}_i > 0$ ,  $f(\tilde{r}_i) < f(-\tilde{r}_i)$ , using  $f(\tilde{r}_i, \tilde{r}_j) = f(\tilde{r}_i | \tilde{r}_j) f(\tilde{r}_j)$ , we can show that, for  $\tilde{r}_i > 0$  and any  $\tilde{r}_j$ ,  $f(\tilde{r}_i, \tilde{r}_j) < f(-\tilde{r}_i, \tilde{r}_j)$ . Similarly, using  $f(-\tilde{r}_i, \tilde{r}_j) = f(\tilde{r}_j | -\tilde{r}_i) f(-\tilde{r}_i)$ , we can show that, for  $\tilde{r}_j > 0$  and any  $\tilde{r}_i$ ,  $f(-\tilde{r}_i, \tilde{r}_j) < f(-\tilde{r}_i, -\tilde{r}_j)$ . Then, the probability of error for bits  $i$  and  $j$ , respectively, is,

$$\begin{aligned} P(i) &= \int_0^\infty \int_{-\tilde{r}_i}^{\tilde{r}_i} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i \\ &= \int_0^\infty \int_0^{\tilde{r}_i} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i + \int_0^\infty \int_{-\tilde{r}_i}^0 f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i \\ P(j) &= \int_{-\infty}^\infty \int_0^{|\tilde{r}_i|} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i \\ &= \int_0^\infty \int_0^{\tilde{r}_i} f(\tilde{r}_i, \tilde{r}_j) d\tilde{r}_j d\tilde{r}_i + \int_0^\infty \int_{-\tilde{r}_i}^0 f(-\tilde{r}_i, -\tilde{r}_j) d\tilde{r}_j d\tilde{r}_i \end{aligned}$$

and thus,  $P(i) \leq P(j)$ .

The second inequality,  $P(i, j) \leq P(i, k)$ , can be proved by assuming conditioning,  $P(i, j) = P(j|i)P(i)$ ,  $P(i, k) = P(k|i)P(i)$ , and  $f(\tilde{r}_i, \tilde{r}_j, \tilde{r}_k) = f(\tilde{r}_j, \tilde{r}_k | \tilde{r}_i) f(\tilde{r}_i)$ , and by using inequality  $P(i) \leq P(j)$ , and following the steps in the first part of the theorem proof.  $\blacksquare$

#### IV. LIST DECODING ALGORITHMS

Using Theorem 1 and Theorem 2 in [3], the original OSD assumes the following list of error patterns,

$$\mathcal{E}_L = \{\mathbf{e}\mathbf{G} : 0 \leq w_H[\mathbf{e}] \leq I, \mathbf{e} \in \mathbb{Z}_2^K\} \quad (10)$$

where  $I$  is the so-called reprocessing order of the OSD, and  $w_H[\mathbf{e}]$  is the Hamming weight of the vector  $\mathbf{e}$ . The list (10) uses a  $K$ -dimensional sphere of radius  $I$  defined about the origin  $\mathbf{0} = (0, \dots, 0)$  in  $\mathbb{Z}_2^K$ . The decoding complexity for the list (10) is  $L = \sum_{l=0}^I \binom{K}{l}$  where  $l$  is referred to as the phase of the order  $I$  reprocessing in [3]. Assuming an AWGN channel, the recommended reprocessing order is  $I = \lceil d_{\min}/4 \rceil \ll K$  where  $\lceil \cdot \rceil$  is the ceiling function. Since the OSD algorithm may become too complex for larger values of  $I$  and  $K$ , a stopping criterion for searching the list  $\mathcal{E}_L$  was developed in [7].

We can identify the following inefficiencies of the original OSD algorithm. First, provided that no stopping nor skipping rules for searching the list of the test error patterns are used, once the MRIPs are found, the ordering of bits within the MRIPs according to their reliabilities becomes irrelevant. Second, whereas the BER performance of the OSD is modestly improving with the reprocessing order  $I$ , the complexity of the OSD increases rapidly with  $I$  [7]. Thus, for given  $K$ , the maximum value of  $I$  is limited by the allowable OSD complexity to achieve a certain target BER performance. We can address the inefficiencies of the original OSD by more carefully exploiting the properties of the probability of bit errors given by Lemma 1 and Theorem 1. Hence, our aim is to construct a well-defined list of the test error patterns *without* considering the stopping and the skipping criteria to search this list.

Recall that the error patterns can be uniquely specified by bits in the MRIPs whereas the bits of the error patterns outside the MRIPs are obtained using the parity check matrix. In order to design a list of the test error patterns independently of the particular generator matrix of the code as well as independently of the particular received sequence, we consider only the bit errors within the MRIPs. Thus, we can assume that, for all error patterns, the bit errors outside the MRIPs affect the value of the metric in (5) equally. More importantly, in order to improve the list decoding complexity and the BER performance trade-off, we consider partitioning of the MRIPs into disjoint segments. This decoding strategy employing the segments of the MRIPs is investigated next.

### A. Segmentation-Based OSD

Assuming  $Q$  disjoint segments of the MRIPs, the error pattern  $\mathbf{e}$  corresponding to the  $K$  MRIPs can be expressed as a concatenation of the  $Q$  error patterns  $\mathbf{e}^{(q)}$  of length  $K_q$  bits,  $q = 1, \dots, Q$ , i.e.,

$$\mathbf{e} = (\mathbf{e}^{(1)}, \dots, \mathbf{e}^{(Q)}) \in \mathbb{Z}_2^K$$

so that  $\sum_{q=1}^Q K_q = K$ , and  $w_H[\mathbf{e}] = w_H[\mathbf{e}^{(1)}] + \dots + w_H[\mathbf{e}^{(Q)}]$ . As indicated by Lemma 1 and Theorem 1, more likely error patterns have smaller Hamming weights and they correct the bit positions with smaller reliabilities. In addition, the decoding complexity given by the total number of error patterns in the list should grow linearly with the number of segments  $Q$ . Consequently, for a small number of segments  $Q$ , it is expected that a good decoding strategy is to decode each segment independently, and then, the final decision is obtained by selecting the best error (correcting) pattern from each of the segments decodings. In this paper, we refine this strategy for  $Q = 2$  segments as a generalization of the conventional OSD having only  $Q = 1$  segment.

Assuming that the two segments of the MRIPs are decoded independently, the list of error patterns can be written as,

$$\mathcal{E}_L = \mathcal{E}_{L_1}^{(1)} \cup \mathcal{E}_{L_2}^{(2)} \quad (11)$$

where  $\mathcal{E}_{L_1}^{(1)}$  and  $\mathcal{E}_{L_2}^{(2)}$  are the lists of error patterns corresponding to the list decoding of the first segment and of the second segment, respectively, and  $L = L_1 + L_2$ . Obviously, fewer errors, and thus, fewer error patterns can be assumed in the shorter segments with larger reliabilities of the received bits. Similarly to the conventional OSD having one segment, for both MRIPs segments, we assume all the error patterns up to the maximum Hamming weight  $I_q$ ,  $q = 1, 2$ . Then, the lists of error patterns in (11) can be defined as,

$$\begin{aligned} \mathcal{E}_{L_1}^{(1)} &= \{(\mathbf{e}, \mathbf{0})\mathbf{G} : 0 \leq w_H[\mathbf{e}] \leq I_1, \mathbf{e} \in \mathbb{Z}_2^{K_1}\} \\ \mathcal{E}_{L_2}^{(2)} &= \{(\mathbf{0}, \mathbf{e})\mathbf{G} : 0 \leq w_H[\mathbf{e}] \leq I_2, \mathbf{e} \in \mathbb{Z}_2^{K_2}\}. \end{aligned} \quad (12)$$

The decoding complexity of the segmentation-based OSD with the lists of error patterns defined in (12) is,

$$L = \sum_{l=0}^{I_1} \binom{K_1}{l} + \sum_{l=0}^{I_2} \binom{K_2}{l}$$

where  $K = K_1 + K_2$ , and we assume  $I_1 \ll K_1$  and  $I_2 \ll K_2$ .

Recall that the original OSD, denoted as  $\text{OSD}(I)$ , has one segment of length  $K$  bits, and that the maximum number of bit errors assumed in this segment is  $I$ . The segmentation-based OSD is denoted as  $\text{OSD}(I_1, I_2)$ , and it is parametrized by the segment length  $K_1$ , and  $K_2$ , and the maximum number of errors  $I_1$  and  $I_2$ , respectively. The segment sizes  $K_1$  and  $K_2$  are chosen empirically to minimize the BER for a given decoding complexity and for a class of codes under consideration. In particular, for systematic block codes of block length  $N < 128$  and of rate  $R \geq 1/2$ , we found that the recommended length of the first segment is,

$$K_1 \approx 0.35K$$

so that the second segment length is  $K_2 = K - K_1$ . The maximum number of bit errors  $I_1$  and  $I_2$  in the two segments are selected to fine-tune the BER performance and the decoding complexity trade-off. For instance, we can obtain the list decoding schemes having the BER performance as well as the decoding complexity between those corresponding to the original decoding schemes  $\text{OSD}(I)$  and  $\text{OSD}(I + 1)$ .

Finally, we note that it is straightforward to develop the skipping criteria for efficient searching of the list of error patterns in the OSD-based decoding schemes. In particular, one can consider the Hamming distances for one or more segments of the MRIPs between the received hard decisions (before the decoding) and the temporary decisions obtained using the test error patterns from the list. If any or all of the Hamming distances are above given thresholds, the test error pattern can be discarded without re-encoding and calculating the corresponding Euclidean distance. For the  $Q = 2$  segments OSD, our empirical results indicate that the thresholds for the first and the second segments should be  $0.35 d_{\min}$  and  $d_{\min}$ , respectively.

### B. Partial-Order Statistics Decoding

The Gauss (or the Gauss-Jordan) elimination employed in the OSD-based decoding algorithms represents a significant portion of the overall implementation complexity. A new row (or a reduced row) echelon form of the generator matrix must be obtained after every permutation  $\lambda''$  until the MRIPs are found. Hence, we can devise a partial-order statistics decoding (POSD) that completely avoids the Gauss elimination, and thus, it further reduces the decoding complexity of the OSD-based decoding. The main idea of the POSD is to order only the first  $K$  received bits according to their reliabilities, so that the generator matrix remains in its systematic form.

The ordering of the first  $K$  received bits in the descending order can improve the coding gain of the segmentation-based OSD. Assuming  $Q = 2$  segments, we use the notation  $\text{POSD}(I_1, I_2)$ . The parameters  $K_1, K_2, I_1$  and  $I_2$  of  $\text{POSD}(I_1, I_2)$  can be optimized similarly as in the case of  $\text{OSD}(I_1, I_2)$  to fine-tune the BER performance versus the implementation complexity. On the other hand, we will show in Section V that the partial ordering (i.e., the ordering of the first  $K$  out of  $N$  received bits) is irrelevant for the OSD decoding having one segment of the MRIPs and using the list of error patterns (10). In this case, the  $\text{POSD}(I)$  decoding can be referred to as the input-sphere decoding  $\text{ISD}(I)$ .

### C. Implementation Complexity

We compare the number of binary operations (BOPS) and the number of floating point operations (FLOPS) required to execute the decoding algorithms proposed in this paper. Assuming a  $(N, K, d_{\min})$  code, the complexity of the OSD and the POSD are given in Table I and Table II. The implementation complexity expressions in Table I for  $\text{OSD}(I)$  are from the reference [3]. For example, the OSD decoding of the BCH code  $(128, 64, 22)$  requires at least 1152 FLOPS and 528448 BOPS to find the MRIPs and to obtain the corresponding equivalent generator matrix in a row echelon form. All this complexity can be completely avoided by assuming the partial ordering in the POSD decoding. The number of the test error patterns is  $L = 2080$  for  $\text{OSD}(2)$ , and  $L = 1177$  for  $\text{OSD}(2, 2)$  with  $K_1 = 21$  and  $K_2 = 43$  whereas the coding gain of  $\text{OSD}(2)$  can be only slightly better than the coding gain of  $\text{OSD}(2, 2)$ ; see, for example, Fig. 4. Hence, the overall complexity of the OSD-based schemes can be substantially reduced by avoiding the Gauss (Gauss-Jordan) elimination.

## V. PERFORMANCE ANALYSIS

Recall that we assume a memoryless communication channel as described in Section II. We derive the probability  $\Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}_L\}$  in (9) that the error pattern  $\hat{\mathbf{c}}_0$  observed at the receiver after transmission of the codeword  $\mathbf{c}_{\text{Tx}} = \mathbf{0}$  is an element of the chosen decoding list  $\mathcal{E}_L$ . The derivation relies on the following generalization of Lemma 3 in [3].

*Lemma 2:* For any ordering of the  $N$  received bits, consider the  $I$  bit positions  $\mathcal{I} \subseteq \{1, 2, \dots, N\}$ , and the  $\binom{I}{I_1}$  subsets  $\mathcal{I}_1 \subseteq \mathcal{I}$  of  $I_1 \leq I \leq N$  bit positions. Then, the total probability of the  $I_1$

bit errors within the  $I$  bits can be calculated as,

$$\sum_{\mathcal{I}_1: |\mathcal{I}_1|=I_1} P(\mathcal{I}_1) = \binom{I}{I_1} p_0^{I_1}$$

where  $p_0$  is the probability of bit error corresponding to the bit positions  $\mathcal{I}$ .

*Proof:* The ordering of the chosen  $I$  bits given by the set  $\mathcal{I}$  is irrelevant since *all* subsets  $\mathcal{I}_1$  of  $I_1$  errors within the  $I$  bits  $\mathcal{I}$  are considered. Consequently, the bit errors in the set  $\mathcal{I}$  can be considered to be independent having the equal probability denoted as  $p_0$ .  $\blacksquare$

Using Lemma 2, we observe that the lists of error patterns (10) and (12) are constructed, so that the ordering of bits within the segments is irrelevant. Then, the bit errors in a given segment can be considered to be conditionally independent. This observation is formulated as the following corollary of Lemma 2.

*Corollary 1:* For the  $\text{OSD}(I)$  and the list of error patterns (10), the bit errors in the MRIPs can be considered as conditionally independent. Similarly, for the  $\text{POSD}(I_1, I_2)$  and the list of error patterns (12), the bit errors in the two segments can be considered as conditionally independent.

Thus, the bit errors in Corollary 1 are independent conditioned on the particular segment being considered as shown next.

Let  $P_0$  be the bit error probability of the MRIPs for the  $\text{OSD}(I)$  decoding. Similarly, let  $P_1$  and  $P_2$  be the bit error probabilities in the first and the second segments of the  $\text{OSD}(I_1, I_2)$  decoding, respectively. Denote the auxiliary variables,  $v_1 = |\tilde{r}_{K_1}|$ ,  $v_2 = |\tilde{r}_{K_1+1}|$ , and  $v_3 = |\tilde{r}_{K+1}|$  of the order statistics (3), and let  $u \equiv |r_i|$ ,  $i = 1, 2, \dots, K$ . Hence, always,  $v_1 \geq v_2$ , and, for simplicity, ignoring the second permutation  $\lambda''$ , also,  $v_2 \geq v_3$ . The probability of bit error  $P_0$  for the MRIPs is calculated as,

$$P_0 = E_u \left[ \int_0^u \frac{f_{v_3}(v)}{1 - F_u(v)} dv \right]$$

where  $E_u[\cdot]$  denotes the expectation w.r.t. (with respect to)  $u$ ,  $f_{v_3}(v)$  is the PDF of the  $(K+1)$ -th order statistic in (3), and  $F_u(v)$  is the cumulative distribution function (CDF) of the magnitude (the absolute value) of the reliability of the received bits (before ordering). Similarly, the probability of bit error  $P_1$  for the first segment is calculated as,

$$P_1 = E_u \left[ \int_0^u \frac{f_{v_2}(v)}{1 - F_u(v)} dv \right]$$

where  $f_{v_2}(v)$  is the PDF of the  $(K_1 + 1)$ -th order statistic in (3). The probability of bit error  $P_2$  for the second segment is calculated as,

$$P_2 = E_u \left[ \int_0^u \int_u^\infty \frac{f_{v_1}(v) f_{v_3}(v')}{(F_u(v) - F_u(v'))(1 - F_{v_1}(v'))} dv dv' \right]$$

where  $f_{v_1}(v)$  and  $F_{v_1}(v')$  is the PDF and the CDF of the  $K_1$ -th order statistic in (3), respectively. The values of the probabilities  $P_0$ ,  $P_1$  and  $P_2$  have to be evaluated by numerical integration. Finally, we use Lemma 2 and substitute the probabilities  $P_0$ ,  $P_1$  and  $P_2$  for  $p_0$  to calculate the probability  $\Pr\{\hat{\mathbf{c}}_0 \in \mathcal{E}_L\}$  of the error patterns in the list  $\mathcal{E}_L$ .

## VI. NUMERICAL EXAMPLES

We use computer simulations to compare the BER performances of the proposed soft-decision decoding schemes. All the block codes considered are linear and systematic.

The BER of the  $(31, 16, 7)$  BCH code over an AWGN channel is shown in Fig. 2 assuming ISD(2) and ISD(3) with  $K = 16$  having 137 and 697 test error patterns, respectively, and assuming POSD(1, 3) and POSD(2, 3) with  $K_1 = 6$  and  $K_2 = 10$  having 183 and 198 test error patterns, respectively. We observe that POSD(1, 3) achieves the same BER as ISD(3) while using much less error patterns which represents the gain of the ordering of the received information bits into two segments. At the BER of  $10^{-4}$ , POSD(1, 3) outperforms ISD(2) by 1.1 dB using approximately 50% more test error patterns. Thus, the POSD(1, 3) decoding provides 2.3 dB coding gain with the small implementation complexity at the expense of 2 dB loss compared to the ML decoding.

Fig. 3 shows the BER of the  $(63, 45, 14)$  BCH code over an AWGN channel. The number of test error patterns for the ISD(2), ISD(3), POSD(1, 3) and OSD(2) decodings are 1036, 15226, 5503 and 1036, respectively. We observe from Fig. 3 that ISD(3) has the same BER as POSD(1, 3) with two segments of  $K_1 = 13$  and  $K_2 = 32$  bits. However, especially for the high rate codes (i.e. of rates greater than  $1/2$ ), one has to also consider the complexity of the Gauss elimination to obtain the row echelon form of the generator matrix for the OSD. For example, the Gauss elimination for the  $(63, 45, 14)$  code requires approximately 20,400 BOPS; cf. Table I.

The BER of the  $(128, 64, 22)$  BCH code over an AWGN channel is shown in Fig. 4 assuming OSD(1) and OSD(2) with  $K = 64$ , and assuming OSD(2, 2) with  $K_1 = 21$  and  $K_2 = 43$ . The number of test error patterns for the OSD(1), OSD(2) and OSD(2, 2) decodings are 64, 2081

and 1179. A truncated union bound of the BER in Fig. 4 is used to indicate the ML performance [14, Ch. 10]. We observe that both OSD(2) and OSD(2, 2) have the same BER performance for the BER values larger than  $10^{-3}$ , and *OSD*(2) outperforms OSD(2, 2) by at most 0.5 dB for the small values of the SNR. Our numerical results indicate that, in general, OSD(2, 2) decoding can achieve approximately the same BER as OSD(2) for small to medium SNR while using about 50% less error patterns. Thus, a slightly smaller coding gain (less than 0.5dB) of OSD(2, 2) in comparison with OSD(2) at larger values of the SNR is well-compensated for by the reduced decoding complexity. More importantly, OSD(2, 2) can trade-off the BER performance and the decoding complexity between those provided by OSD(1) and OSD(2), especially at larger values of SNR.

The BER of the (31, 16, 7) BCH code over a fast Rayleigh fading channel is shown in Fig. 5. We assume the same decoding schemes as in Fig. 2. The POSD(1, 3) decoding with 183 error patterns achieves the coding gain of 17 dB over an uncoded system, the coding gain of 4 dB over ISD(2) with 137 error patterns, and it has the same BER as OSD(3) with 697 error patterns. The BER of the high rate BCH code (64, 57, 4) over a fast Rayleigh channel is shown in Fig. 6. In this case, the number of test error patterns for the ISD(2), ISD(3), POSD(2, 3) and OSD(2) decoding is 1654, 30914, 8685 and 1654, respectively. We observe that, for small to medium SNR, POSD(2, 3) which does not require the Gauss elimination (corresponding to approximately 3,000 BOPS) outperforms OSD(2) by 1dB whereas, for large SNR values, these two decoding schemes achieve approximately the same BER performance.

## VII. CONCLUSIONS

Low-complexity soft-decision decoding techniques employing a list of the test error patterns for linear binary block codes of small to medium block length were investigated. The optimum and suboptimum construction of the list of error patterns was developed. Some properties of the joint probability of error of the received bits after ordering were derived. The original OSD algorithm was generalized by assuming a segmentation of the MRIPs. The segmentation of the MRIPs was shown to overcome several drawbacks of the original OSD and to enable flexibility for devising new decoding strategies. The decoding complexity of the OSD-based decoding algorithms was reduced further by avoiding the Gauss (or the Gauss Jordan) elimination using the partial ordering of the received bits in the POSD decoding. The performance analysis was

concerned with the problem of finding the probability of the test error patterns contained in the decoding list. The BER performance and the decoding complexity of the proposed decoding techniques were compared by extensive computer simulations. Numerical examples demonstrated excellent flexibility of the proposed decoding schemes to trade-off the BER performance and the decoding complexity. In some cases, both the BER performance as well as the decoding complexity of the segmentation-based OSD were found to be improved compared to the original OSD.

## APPENDIX

We derive the probabilities  $P_0$ ,  $P_1$  and  $P_2$  in Section V. Without loss of generality, we assume that the all-ones codeword was transmitted, i.e.,  $x_i = -1$  for  $\forall i$ . Then, after ordering, the  $i$ -th received bit,  $i = 1, 2, \dots, N$ , is in error, provided that  $\tilde{r}_i > 0$ . The probability of bit error  $P_0$  for the MRIPs is obtained as,

$$P_0 = \int_0^\infty \int_0^\infty f_u(u|u \geq v_3) f_{v_3}(v_3) dv_3 du$$

where the conditional PDF [16],

$$f_u(u|u \geq v_3) = \begin{cases} \frac{f_u(u)}{1-F_u(v_3)} & u \geq v_3 \\ 0 & u < v_3 \end{cases}$$

and  $f_u(u)$  and  $F_u(v_3)$  are the PDF and the CDF of the reliability of the received bits, respectively, so that,

$$P_0 = \int_0^\infty f_u(u) \int_0^u \frac{f_{v_3}(v_3)}{1-F_u(v_3)} dv_3 du.$$

Similarly, the probability of bit error  $P_1$  for the first segment is calculated as,

$$\begin{aligned} P_1 &= \int_0^\infty \int_0^\infty f_u(u|u \geq v_2) f_{v_2}(v_2) dv_2 du \\ &= \int_0^\infty f_u(u) \int_0^u \frac{f_{v_2}(v_2)}{1-F_u(v_2)} dv_2 du. \end{aligned}$$

The probability of bit error  $P_2$  for the second segment is calculated as,

$$P_2 = \int_0^\infty \int_0^\infty \int_0^\infty f_u(u|v_1 \geq u \geq v_3) f_{v_1, v_3}(v_1, v_3) dv_1 dv_3 du$$

where the conditional PDF,

$$f_u(u|v_1 \geq u \geq v_3) = \begin{cases} \frac{f_u(u)}{F_u(v_1)-F_u(v_3)} & v_1 \geq u \geq v_3 \\ 0 & \text{otherwise} \end{cases}$$

and the joint PDF of the order statistics  $v_1 \geq v_3$  is,

$$f_{v_1, v_3}(v_1, v_3) = \begin{cases} \frac{f_{v_1}(v_1)}{1-F_{v_1}(v_3)} f_{v_3}(v_3) & v_1 \geq v_3 \\ 0 & v_1 < v_3 \end{cases}$$

and thus,

$$P_2 = \int_0^\infty f_u(u) \int_0^u \int_u^\infty \frac{f_{v_1}(v) f_{v_3}(v')}{(F_u(v) - F_u(v'))(1 - F_{v_1}(v'))} dv dv' du.$$

## REFERENCES

- [1] H. Yagi, "A study on complexity reduction of the reliability-based maximum likelihood decoding algorithm for block codes," PhD dissertation, Waseda University, 2005.
- [2] B. Dorsch, "A decoding algorithm for binary block codes and  $J$ -ary output channels," *IEEE Trans. Inf. Theory*, vol. 20, pp. 391-394, 1974.
- [3] M. P. C. Fossorier and S. Lin, "Soft-decision decoding of linear block codes based on ordered statistics," *IEEE Trans. Inf. Theory*, vol. 41, pp. 1379-1396. Sept. 1995.
- [4] D. Gazelle and J. Snyders, "Reliability-based code-search algorithms for maximum-likelihood decoding of block codes," *IEEE Trans. Inf. Theory*, vol. 43, pp. 239-249, Jan. 1997.
- [5] A. Kabat, F. Guilloud and R. Pyndiah, "New approach to order statistics decoding of long linear block codes," in *Proc. Globecom*, pp. 1467-1471, Nov. 2007.
- [6] H. Yagi, T. Matsushima and S. Hirasawa, "Fast algorithm for generating candidate codewords in reliability-based maximum likelihood decoding," *IEICE Trans. Fundamentals*, vol. E89-A, pp. 2676-2683, Oct. 2006.
- [7] M. Fossorier and S. Lin, "Computationally efficient soft decision decoding of linear block codes based on ordered statistics," *IEEE Trans. Inf. Theory*, vol. 42, pp. 738-750, May 1996.
- [8] M. Fossorier, "Reliability-based soft-decision decoding with iterative information set reduction," *IEEE Trans. Inf. Theory*, vol. 48, no. 12, pp. 3101-3106, Dec. 2002.
- [9] W. Jin and M. Fossorier, "Reliability based soft decision decoding with multiple biases," *IEEE Trans. Inf. Theory*, vol. 53, no. 1, pp. 105-119, Jan. 2007.
- [10] A. Valembois and M. Fossorier, "Sort-and-match algorithm for soft-decision decoding," *IEEE Trans. Inf. Theory*, vol. 45, pp. 2333-2338, Nov. 1999.
- [11] A. Valembois and M. Fossorier, "Box and match techniques applied to soft decision decoding," *IEEE Trans. Inf. Theory*, vol. 50, no. 5, pp. 796-810, May 2004.
- [12] M. El-Khamy, H. Vialko, B. Hassibi and R. McEliece, "Performance of sphere decoding of block codes," *IEEE Trans. Comms.*, vol. 57, pp. 2940-2950, Oct. 2009.
- [13] H. Vikalo and B. Hassibi, "On joint detection and decoding of linear block codes on Gaussian vector channels," *IEEE Trans. Signal Proc.*, vol. 54, pp. 3330-3342, Sep. 2006.
- [14] S. Benedetto and E. Biglieri, *Principles of Digital Transmission With Wireless Applications*, Kluwer Academic, 1999.
- [15] S. Lin and D. J. Costello, *Error Control Coding: Fundamentals and Applications*, Prentice-Hall, 1983.
- [16] A. Papoulis and S. U. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th Ed., McGraw-Hill, 2002.

TABLE I  
IMPLEMENTATION COMPLEXITY OF THE OSD AND THE POSD

OSD( $I_1$ ) and OSD( $I_1, I_2$ )	
operation	complexity
$\mathbf{r}$	$2N$ FLOPS
$\tilde{\mathbf{r}}'$	$N \log_2(N)$ FLOPS
Gauss el. $\mathbf{G}'$	$N \min(K, N - K)^2$ BOPS
$\tilde{\mathbf{r}}''$	$K + K(N - K)$ BOPS
POSD( $I_1$ ) $\equiv$ ISD( $I_1$ )	
operation	complexity
$\mathbf{r}$	$2N$ FLOPS
$\tilde{\mathbf{r}}'$	0 BOPS
POSD( $I_1, I_2$ )	
operation	complexity
$\mathbf{r}$	$2N$ FLOPS
$\tilde{\mathbf{r}}'$	$K \log_2(K)$ FLOPS

TABLE II  
DECODING LIST SIZES FOR THE OSD AND THE POSD

OSD( $I$ )	$\sum_{l=0}^I \binom{K}{l}$
OSD( $I_1, I_2$ )	$\sum_{l=0}^{I_1} \binom{K_1}{l} + \sum_{l=0}^{I_2} \binom{K_2}{l}$
POSD( $I$ ) $\equiv$ ISD( $I$ )	$\sum_{l=0}^I \binom{K}{l}$
POSD( $I_1, I_2$ )	$\sum_{l=0}^{I_1} \binom{K_1}{l} + \sum_{l=0}^{I_2} \binom{K_2}{l}$

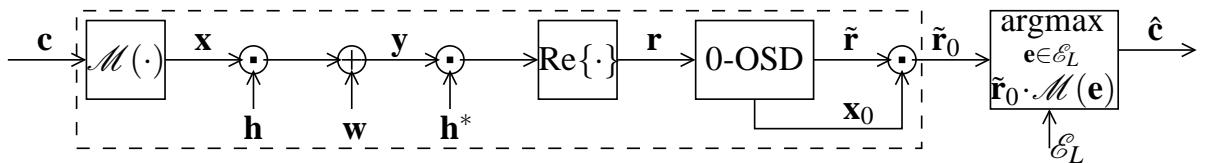


Fig. 1. The system model and an equivalent vector channel with the binary vector input  $\mathbf{c}$  and the vector soft-output  $\tilde{\mathbf{r}}_0$ .

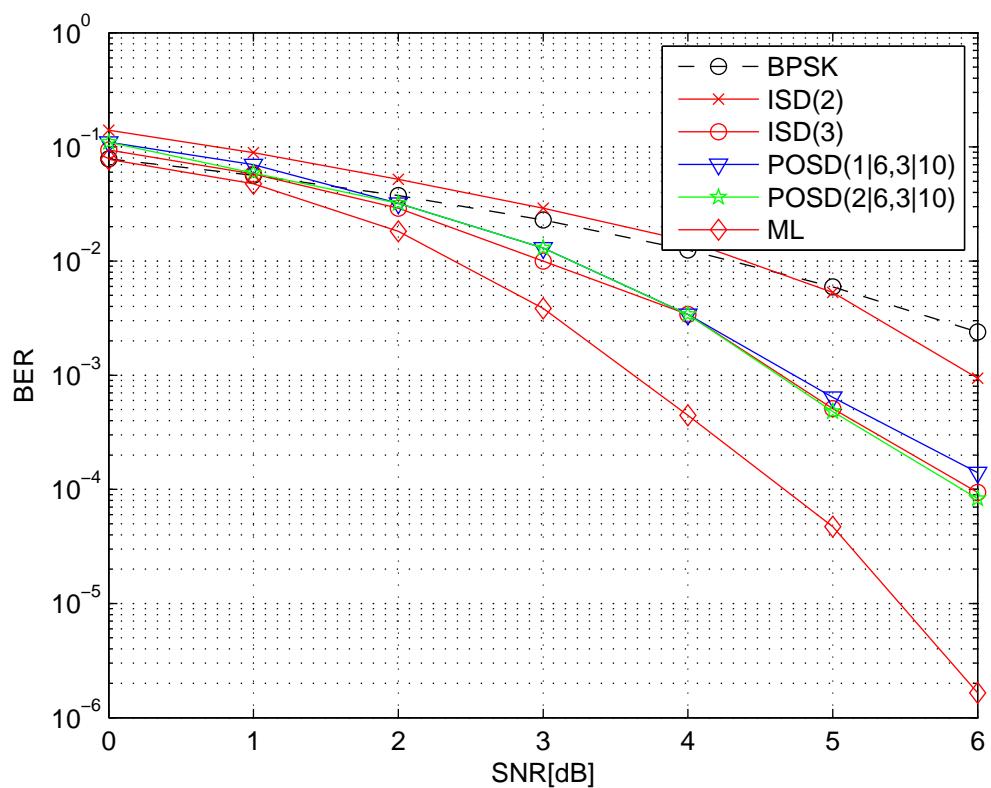


Fig. 2. The BER of the  $(31, 16, 7)$  BCH code over an AWGN channel.

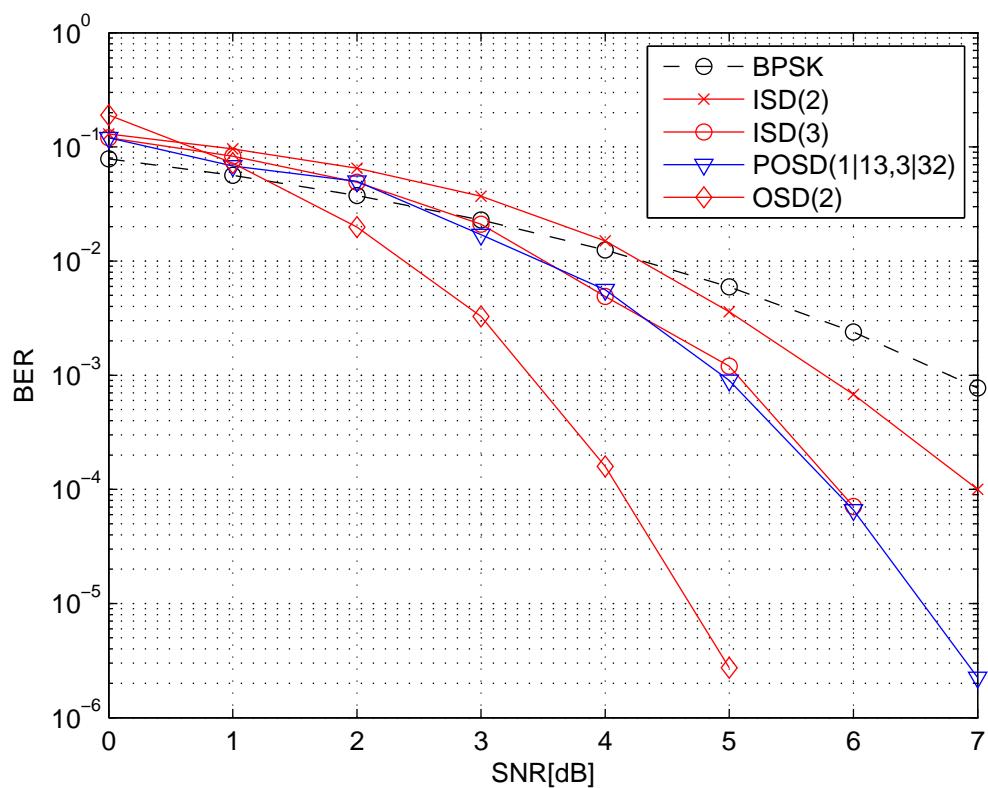


Fig. 3. The BER of the  $(63, 45, 14)$  BCH code over an AWGN channel.

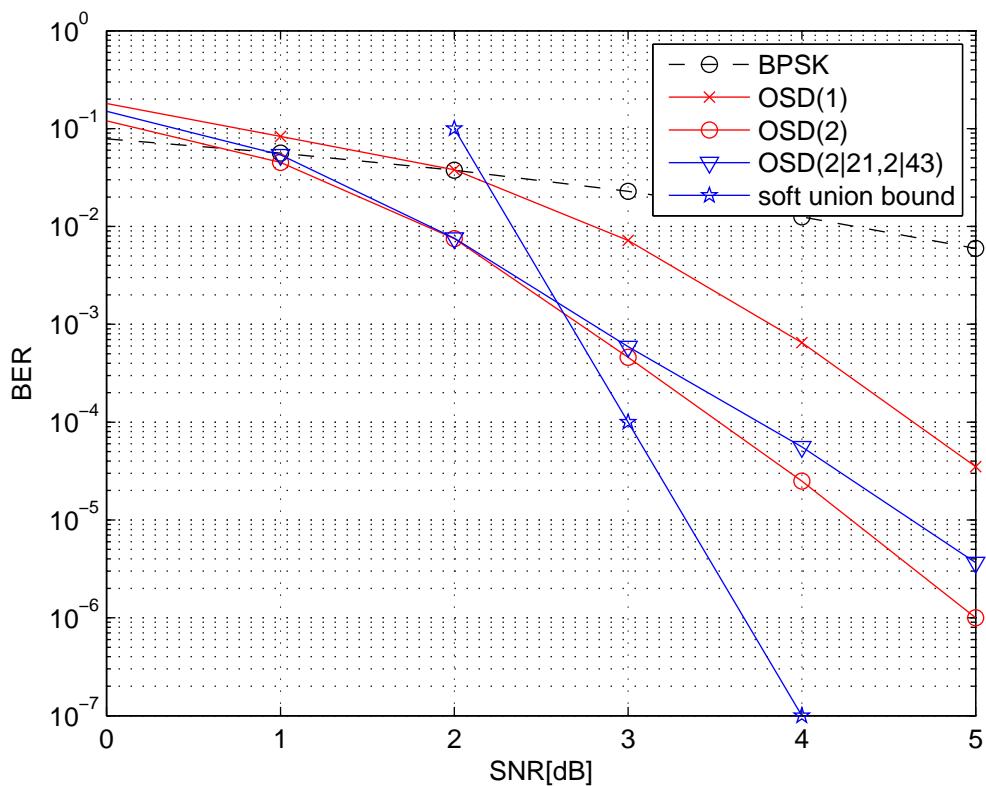


Fig. 4. The BER of the  $(128, 64, 22)$  BCH code over an AWGN channel.

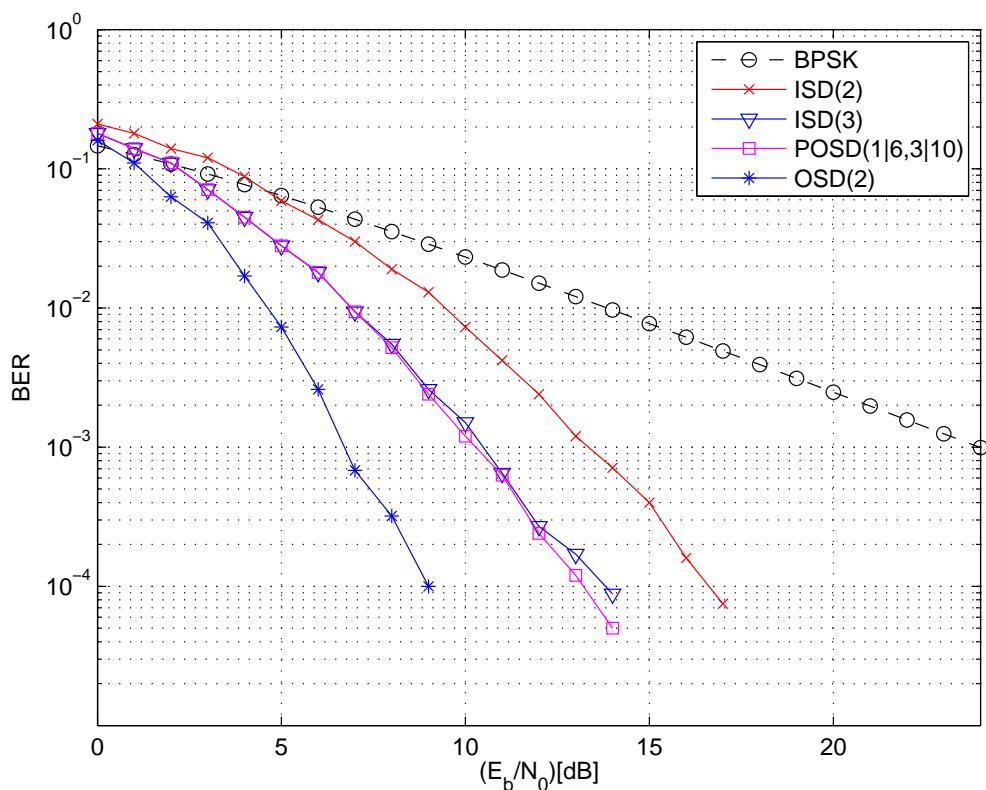


Fig. 5. The BER of the  $(31, 16, 7)$  BCH code over a Rayleigh fading channel.

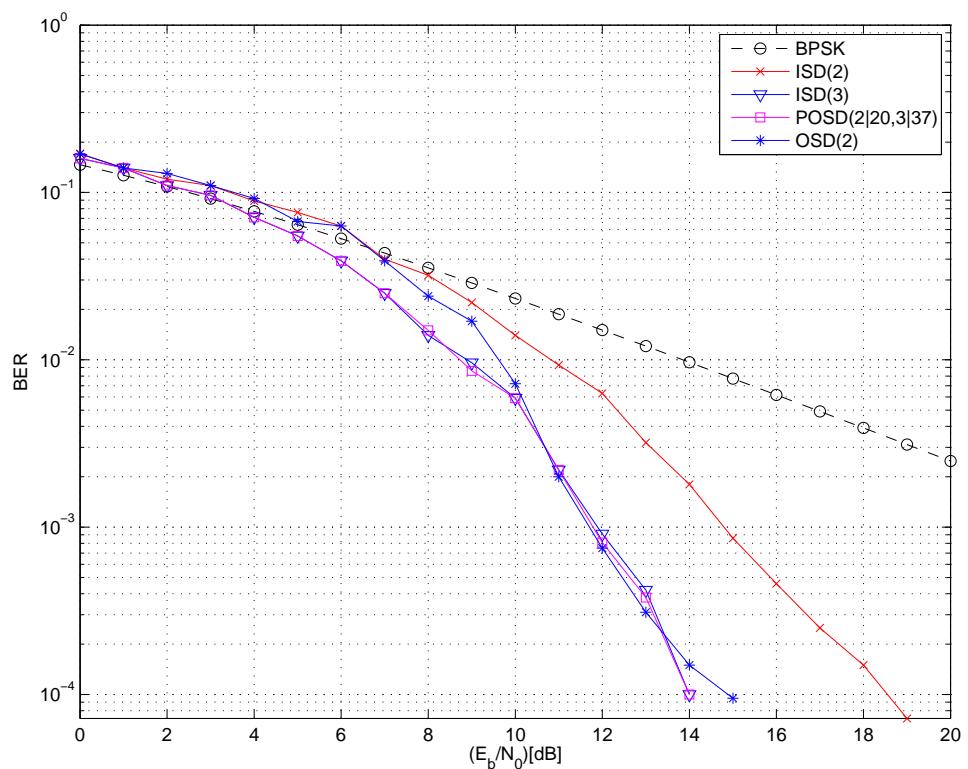


Fig. 6. The BER of the  $(64, 57, 4)$  BCH coded over a Rayleigh fading channel.